

REMARKS

The enclosed is responsive to the Examiner's Office Action mailed on March 27, 2008. At the time the Examiner mailed the Office Action claims 1-36 were pending. By way of the present response the Applicants have: 1) amended claims 1, 13, 18, 26 and 34; 2) added no new claims; and 3) canceled claims 9-11, 23 and 31. As such, claims 1-8, 12-22, 24-30 and 32-36 are now pending. The Applicants respectfully request reconsideration of the present application and the allowance of all claims now represented.

Claim Rejections

35 U.S.C. 103(a) Rejections

The Examiner rejected claims 1-36 under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent 6,539,501 (hereinafter "Edwards") in view of U.S. Patent 6,470,388 (hereinafter "Niemi"). In response, the Applicants have amended independent claims 1, 18 and 26 to respectively recite (emphasis added):

1. An integrated tracing and logging system employed within a system of computers interconnected through a network comprising:
 - a tracing module associated with specified program code regions of an application, the tracing module to receive and process tracing method calls generated by the application when the specified program code regions are executed, the tracing module being an instance of a first sub-class of a class; and
 - a logging module associated with specified categories of the system, the logging module to receive and process logging method calls from components associated with the categories, the logging module being an instance of a second sub-class of the class, the first sub-class being different than the second sub-class;

a trace filter to filter the tracing method calls according to a specified trace message filtering policy wherein the specified trace message filtering policy comprises blocking a trace message from a first output destination but allowing the trace message to be sent to a second output destination.
18. An integrated logging and tracing system employed within a system of computers interconnected through a network comprising:
 - integrated logging and tracing means, the tracing means associated with specified program code regions of an application, the tracing

means to receive and process tracing method calls generated by the application when the specified program code regions are executed; and

the logging means associated with specified categories of the system, the logging means to receive and process logging method calls from components associated with the categories, the means and tracing means being respective instances of different sub-classes of the same class;

filtering means to filter the tracing method calls and the logging method calls according to a specified trace message filtering policy and logging message filtering policy, respectively, the filtering means preventing the calls from reaching respective trace files and log files.

26. A method employed within a system that includes a network comprising:

defining a class hierarchy comprising a controller class, a tracing sub-class, and a logging sub-class, wherein the controller class is a parent class to the tracing sub-class and the logging sub-class;

creating an instance of the tracing sub-class associated with specified program code regions of an application, the tracing instance to receive and process tracing method calls generated by the application when the specified program code regions are executed; and

creating an instance of the logging sub-class associated with specified categories of the system, the logging instance to receive and process logging method calls from components associated with the categories;

filtering the tracing method calls and the logging method calls according to a specified trace message filtering policy and logging message filtering policy, respectively, the filtering preventing the calls from reaching respective trace files and log files.

Thus, the Applicant has amended claim 1 to recite a filtering policy that blocks a message from reaching a first destination but permits the message to reach a second destination. The Applicant has also amended claims 18 and 26 to recite respective trace message and logging message filtering policies (independent claim 34 has also been amended similarly to claim 26).

Support for these amendments may be found at least in paragraph [0023] of the Applicant's specification which states (in reference to Fig. 2 of the Applicant's specification which is also provided further below)(emphasis added):

Before (or after) evaluating the trace/log message based on severity, filters 212 associated with the controller 200 may filter the messages based on predefined filtering criteria. By way of example, if a particular controller 200 is capable of writing to both a file and a console, a filter 212 may be assigned to filter messages directed to the file, thereby limiting output to the console only. Thus, using filters, a finer

granularity of log controller 200 output may be defined, using variables other than merely severity. As illustrated in **Figure 2**, filters may be associated with particular log controllers 200 and/or with specific output destinations 214 (e.g., specific log/trace files). As mentioned above, filters may be associated with both controllers 200 and/or output destinations 214 to further restrict or alter the output tracing and logging behavior. In one embodiment, multiple filters having different filtering criteria may be allocated to each controller 200 and/or destination 214.

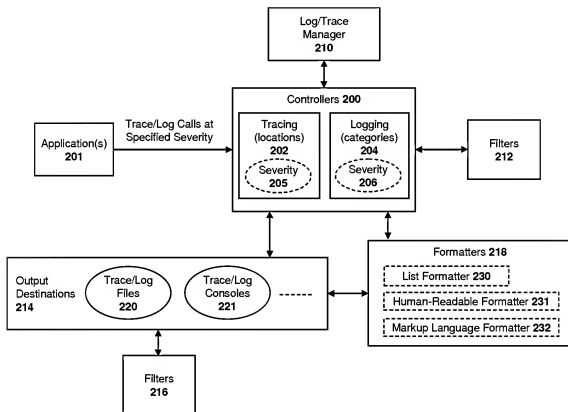


Fig. 2

The Applicant's amendment to claim 1 is based on originally filed dependent claim 9. The amendments to claims 18 and 26 are based on originally filed dependent claims 23 and 31. In the outstanding Office Action the Examiner reasoned that the subject matter of claims 9, 23 and 31 were disclosed by col. 3, line 48 to col. 5, line 48 of Edwards. See, Examiner's Office Action mailed 3/27/08, pgs. 4 and 6.

The Applicant respectfully disagrees. Edwards does not appear to contemplate the specific filtering functions now claimed by the Applicant. That is, Edwards does not appear to disclose a filtering policy that blocks a message from reaching a first destination but permits the message to reach a second destination (as now recited in claim 1) nor does Edwards appear to disclose respective trace message and logging message filtering policies (as now recited in claims 18, 26 and 34).

As the Applicant understand the disclosure of Edwards, the only disclosure of Edwards that could reasonably construed as pertaining to a form of filtering is as follows:

FIG. 2 illustrates an example of parameters that may be used with the log method trace statements. A unique ID 20 provides an identifier for a group of which the log method is associated. For instance, a Java applet program may be comprised of different components, each performing some particular defined function or category of functions within the applet. In such case, the software developer may assign a unique ID to each component of the applet, such that the log methods having the unique ID associated with a particular component would be used to trace the execution of those Java statements related to that particular component. A severity parameter 22 indicates whether the log method is for an error message, warning or information. For instance, if the software developer is inserting the log method trace statements into the source code 6 to debug an error, then the error severity level may be used. A message parameter 24 allows the user to specify a particular message to display in the log file 14.

The preferred log method statement provides a straightforward and easy-to use technique for a software developer to insert trace statements into the source code 6. Although the log method only requires three parameters that are readily specified, the log method calls are capable of causing even further trace information to be written to the log file 14, such as the class and method from which the log method trace was called. However, the software developer need not specify such additional information in the log method calls, as such information is determined by the logger daemon 12 during the process of executing the object code 10.

In preferred embodiments, the software developer would also specify two environmental variables to process the log method trace statements, a TraceOn and groupNumbers variables. TraceOn is a boolean value indicating whether the log method trace statements should be processed. The groupNumbers variable indicates the unique ID of log method trace statements that should be processed. In this

way, the software developer can specify whether to even process log method calls and/or the group of log methods that will be considered.

As the Applicant understands the disclosure of Edwards, all messages that are entered into queue 16 are entered into log file 14. Thus, none of the "filtering" that may be accomplished with the "uniqueID", "TraceOn" or "groupNumbers" variables of Edwards are capable of preventing a message from reaching a first destination but permitting that same message to reach a second destination. Thus Edwards is not capable of disclosing the subject matter that has been added to independent claim 1.

Moreover, with respect to independent claims 18, 26 and 34, again, Edwards does not specify separate "trace files and log files". Instead, Edwards only discloses that all processed messages are entered in same queue 16 and log file 14. Therefore Edwards is not capable of disclosing the subject matter that has been added to independent claims 18, 26 and 34.

The Neimi reference does not appear to cure the deficiencies of Edwards. Neimi discloses a "centralized logging facility" 236 in Figs. 2 and 5. Neimi's discussion of the centralized logging facility - which is substantial with respect to Fig. 5 of Neimi - does not appear to disclose an architecture designed to prevent storage of a message at one destination while permitting storage of the same message at another destination. Likewise, Neimi's centralized logging facility does not appear to include separate trace files and log files. Therefore the Applicant respectfully submits that that independent claims as now presented are allowable over the Examiner's cited combination.

In light of the comments above, the Applicants respectfully request the allowance of all claims.

CONCLUSION

Applicant respectfully submits that all rejections have been overcome and that all pending claims are in condition for allowance.

If there are any additional charges, please charge them to our Deposit Account Number 02-2666. If a telephone conference would facilitate the prosecution of this application, the Examiner is invited to contact Thomas C. Webster at (408) 720-8300.

Respectfully submitted,
BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Date: 7/28/08 /Thomas C. Webster/
Thomas C. Webster
Reg. No.: 46,154

1279 Oakmead Parkway
Sunnyvale, CA 94085-4040
(408) 720-8300